Univerzitet u Nišu

Elektronski fakultet

Katedra za računarstvo

Seminarski rad

**Interna struktura i organizacija skladišta podataka kod MS SQL Server baze podataka**

****

**Student: Profesor:**

Aleksandar Mladenovic, 883 Doc. Dr Aleksandar Stanimirović

April 2021.

Sadržaj

[1. Uvod 3](#_Toc69325419)

[2. Struktura I organizacija skladišta 4](#_Toc69325420)

[2.1. Hijerarhijska struktura memorije 4](#_Toc69325421)

[2.2. Organizacija na disku 5](#_Toc69325422)

[2.2.1. Zapisi 5](#_Toc69325423)

[2.2.2. Datoteke 5](#_Toc69325424)

[2.2.3. Stranice 6](#_Toc69325425)

[2.3. Organizacija datoteka 7](#_Toc69325426)

[3. MS SQL Server 8](#_Toc69325427)

[3.1. Organizacija skladišta 8](#_Toc69325428)

[3.2. Primer organizacije 9](#_Toc69325429)

[4. Zaključak 11](#_Toc69325430)

[5. Literatura 12](#_Toc69325431)

# Uvod

Termin „baze podataka“ se odnosi na skup podataka koji su na određeni način povezani i organizovani. Pristup tim podacima omogućavaju Sistemi za upravljanje bazama podataka, softvera koji pruža korisnicima interakciju sa jednom ili više baza podataka, kao i razne funkcije od unosa, skladištenja i pronalaženja informacija do upravljanja načina na koji su informacije organizovane. U zavisnosti od načina organizacije i manipulacije nad podacima, postoje različiti tipovi baza podataka kao što su: Relacione, Objektno orjentisane, Distibuirane, NoSQL, Graf,...

Još od 50tih godina prošlog veka, za skladištenje i obradu podataka koriste se računari. Primarna memorija računara omogućava samo privremeno skladištenje, dok se za trajno skladištenje koristi sekundarna memorija koja obuhvata fizički nivo arhitekture baze podataka. Ona sadrži i sve potrebne informacije za efikasno učitavanje na primarnoj memoriji kada je to potrebno. Prvobitni sistemi su bili sekvencijalnog tipa što znači da su sukcesivni zapisi morali biti obrađivanui u redosledu u kojem su snimljeni. Kada su razvijeni uređaji za skladištenje sa diretnim pristupom koji koriste tehniku indeksiranja čime se omogućilo lociranje slučano biranih zapisa. Skladišta su uobičajno orjentisana na redove ali postoje baze orjentisane na kolone i korelacije. Jedna od takvih baza koja je orjentisana na redove je i Microsoft SQL Server, koja spada u relacione baze razvijena od strane Microsofta. U daljem tekstu biće više reči uopšteno o skladištenju podataka i njenoj primeni kod MS SQL Servera baze podataka.

# Struktura I organizacija skladišta

Kolekcije podataka koje čine bazu podataka mora fizički da se čuva na nekom od računarskih medijuma za skladištenje. Na taj način softver može da pristupi i preuzme podatke, ažurira i obradi ih po poptrebi. Medijumi za računarsko skladištenje su hijerarhijski organizovani i mogu se podeli na dve kategorije: Primatno skladište i Sekundarno i tercijarno skladište.

**Primarno skladište** predstavlja medijume kojima direktno upraljva centralna procesorska jedinica (CPU), kao što su glavna memorija računara, i medijume kao što su manje ali brže keš memorije. Glavni zadatak ovog skladišta je brz pristup podacima, ali ograničenog kapaciteta za skladištenje. Iako poslednjih godina kapacitet glavne memorije neprestano raste, oni su i dalje preskupi i manjeg su kapaciteta od sekundarnih i tercijalnih medijuma za skladištenje.

**Sekundarno i tercijarno skladište** čine magnetni diskovi, optički diskovi i trake, gde hard diskovi pripadaju sekundarnom a prenosivi mediji kao optički disk i trake pripadaju tercijarnom. Svima njima je zajedničko da imaju veliku kapacitivnost memorije, jeftiniji su ali im je pristup podacima dosta sporiji od medija primarnog skladišta. Takođe, CPU ne može direktno da pristupa i obrađuje podatke, već se podaci prvo moraju prebaciti u primarno skladište kako bi CPU mogao da ih obrađuje.

## Hijerarhijska struktura memorije

U savremenim računarskim sistemima, podaci se nalaze i prenose kroz hijerarhijski sistem medija za skladištenje. Po brzini memorije, počev od nivoa primarnog skladišta, na vrhu strukture se nalazi keš memorija, statička RAM (Random Access Memory) memorija. Ona se najčešće koristi za ubrzavanje izvršenja instrukcija programa od strane CPU. Posle nje sledi memorija DRAM (Dynamic RAM), koja se još naziva i glavna memorija. Čuva programske instrukcije i podatke za CPU. Njegova prednost je niska cena, koja postepeno opada a nedostatak je nestabilonost i manja brzina u odnosu na RAM. Na nivou sekundarnog i tercijarnog skladišta, po hijerarhijskom redosledu idu magnetni diskovi, optički diskovi kao masovna skaldišta u obliku CD-ROM i DVD uređaja i na kraju trake kao najjeftiniji i koji se koriste za arhiviranje i rezervno skladištenje podataka.

## Organizacija na disku

Podaci su na disku organizovani u **blokove** (**stranice**, *pages*) fiksne veličine (512 do 8192 bajtova), u bazi se skladište u formi **slogova** (records), a kolekcija slogova predstavlja **datoteku** ili fajl (file).

## Zapisi

Podaci se na disku obično čuvaju u obliku zapisa. Svaki zapis se sastoji od kolekcije povezanih vrednosti podataka, gde je svaka vrednost formirana od jednog ili više bajtova i odgovara određenom polju zapisa. Zapisi obično opisuju entitete i njihove atribute. Kolekcija imena polja i njihovih odgovarajućih tipova podataka predstavlja tip zapisa ili definiciju formata zapisa. Tip podatka povezan sa svakim poljem, određuje tipove vrednosti koje polje može uzeti. Tip podatka polja je obično jedan od standardnih tipova podataka koji se koristi u programiranju. Tu spadaju numerički, niz znakova, logički, a ponekad i posebno kodirani tipovi podataka o datumu i vremenu. U nekim aplikacijama, može se javiti potreba za skladištenjem podataka koji se sastoje od velikih nestuktuiranih objekata koji predstavljaju slike, digitalizovane video ili audio tokove i slobodne tekstove. Oni se nazivaju BLOB (binarni veliki objekti). BLOB podatak se obično čuva odvojeno od zapisa u skupu blokova diska, a u zapis je uključen i pokazivač na BLOB.

## Datoteke

Fajl ili datoteka je niz zapisa. U većini slučajeva su svi zapisi u datoteci zapravo ista vrsta zapisa. Kada svaki zapis u datoteci ima apsolutno istu veličinu u bajtovima znači da njega čine zapisi fiksne dužine. Kada su različiti zapisi u datoteci različite veličine kaže se da je datoteka sastavljena od zapisa različite dužine. Datoteka može imati zapise promenljive dužine iz više razloga:

* Zapisi (records) datoteka su istog tipa zapisa , ali jedno ili više polja su različite veličine (polja promenljive dužine).
* Zapisi datoteka su istog tipa zapisa medjutim, jedno ili više polja mogu imati više različitih vrednosti za pojedinačne zapise. Takvo polje se naziva ponavljajuće polje (arepeating field), dok grupa vrednosti za polje se naziva vrlo često ponavljajuća grupa (repeating group).
* Zapisi datoteke su istog tipa zapisa, ali jedno ili više polja je neobavezno (opciono), ono može imati vrednost samo za neke ali ne i za sve zapise datoteke (optional fields).
* Datoteka sadrži zapise različite vrste a samim tim i različite veličine (mešovite datoteke). To bi se dogodilo kada bi se različite vrste zapisa grupisali na blokove diskova.

Kod zapisa fiksne dužine svaki zapis ima ista polja, a dužina polja su fiksna, tako da sistem može identifikovati položaj početnog bajta svakog polja u odnosu na početni položaj tog zapisa. To olakšava pronalaženje vrednosti polja programima koji pristupaju datotekama.

## Stranice

Posmatrajući teoretski SQL server bi mogao vrlo jednostavno da uskladišti milijardu zapisa u jednoj ogromnoj datoteci, ali bi to predstavljalo nered za upravljanje. Zato on umesto toga, organizuje i čuva zapise u manjim jedinicama podataka koje se poznate kao stranice (pages). Stranice su i ujedno najmanje jedinice podataka koje će SQL server keširati u memoriju kojim upravlja buffer mendzer. U zavisnosti od onoga šta skladište postoje različite vrste stranica, koje skladište:

* zapise podataka,
* indeksne zapise,
* metapodatke različitih vrsta.

Svima je zajedničko jedno, njihova struktura. Stranica je uvek tačna 8 KB (8192 bajta) i sadrži dva glavna odeljka: zaglavlje i telo.

Zaglavlje ima fiksnu veličinhu od 96 bajtova i ima isti sadržaj i format bez obzira na tip stranice. Sadrži informacije kao što su: broj slobodnog prostora u telu, broj zapisa koji se čuvaju u telu, broj objekta kome stranica pripada i u indeksu, stranica koje joj prethode i nasleđuju je.

Telo zauzima preostalih 8096 bajtova. Na samom kraju tela nalazi se odeljak koji je poznat kao niz pomeranja zapisa, on predstavlja niz dvobajtnih vrednosti koje SQL Server čita obrnuto od samog kraja stranice.

Zaglavlje sadrži polje koje definiše broj slotova koji su prisutni u polju pomeranja zapisa, a samim tim toliko dvobajtnih vrednosti može da pročita SQL server. Svaki slot u polju pomeranja zapisa ukazuje na indeks bajtova na stranici na kojoj započinje zapis. Samim tim poslednji zapis na stranici može biti fizički i prvi zapis. Obično nam se može ukazati da prvi slot niza pomeranja zapisa koji je smešten u poslednja dva bajta stranice , pokazuje na prvi zapis sačuvan sa indeksom bajtova 96 koji je i na samom početku tela, odmah nakon zaglavlja.

## Organizacija datoteka

Organizacija fajlova podrazumeva način na koji su zapisi i stranice poređane na spoljnom medijumu i način na koji su međusobno povezani. Vrlo često prilikom korišćenja baza, postoje neki uslovi upita koji se često koriste. Isto tako su sa jedne strane neki fajlovi često podložni update operacijama kao što su insert, delete itd. a sa druge strane su drugi više statični. Organizacija fajlova a pre svega dobra treba doprineti tome da se operacije kojima se često služimo i primenjujemo nad fajlovima budu izvršavane efikasnije i brže.

Alternative koje se koriste prilikom organizacije datoteka :

1. **Heap Files** - ovakva organizacija podrazumeva da se slogovi se dodaju u fajl po redosledu njihovog dodavanja u bazu, i to zapravo znači da se novi slogovi dodaju na kraj fajla. Veliki broj baza koristi ovakvu organizaciju u kombinaciji sa indeksima, oni se koriste radi efikasnije pretrage. Heap organizacija je značajna jer omogućava brzo dodavanje podataka, i to podrazumeva operacije kopiranja zadnjeg bloka fajla u glavnu memoriju, dodavanje zapisa u blok, upis bloka na disk. Međutim, postoji i mana ovakve organizacije koja podrazumeva neefikasnu pretragu.

2. **Sorted Files** - Organizacija ovog tipa se dobija tako što se zapisi u fajlu sortiraju na osnovu vrednosti nekog polja (ordering filed). Ona olakšava pretragu na osnovu ordering field vrednosti (primenjuje se binary search tehnika) , kao i pribavljanje svih ili grupe sortiranih zapisa, takođe na osnovu ordering field-a. Operacija kao što je dodavanje i brisanje je složenija u odnosu na heap organizaciju, zato što se mora očuvati uređenost. Isto tako kada se zahteva pretraga na osnovu nekog polja koje nije ordering filed, sortirani fajl ne obezbeđuje nikakve prednosti tj. mora se primeniti linearna pretraga.

3. **Hashed files** - Ova organizacija se zasniva na tehnici hashing-a, koja omogućava brz pristup na osnovu jednakosti. U okviru ove organizacije polje na osnovu koga se vrši poređenje se naziva hash field, takodje u većini slučajeva hash field je istovremeno i ključ (hash key).Ukoliko je potrebno pribaviti, dodati, izvršiti update, ili obrisati zapis, hash funkcija određuje adresu traženog bloka, i ona to vrši koristeći se hash filed vrednostima. Hash funkcija raspoređuje zapise uniformno po adresnom prostoru, pa se zato korišćenjem ovakavim načinom skladištenja zapisi smeštaju na random memorijskim lokacijama.

4. **Indexes** - To je pomoćna struktura podataka koja nam pomaže dase ubrzaju pribavljanja zapisa kod specifičnih vrsta pretraga. Ova ideja je bazirana na indeksima na kraju knjiga, kod kojih se pored svakog termina nalaze strane na kojima se konkretni termin pojavljuje. To znači da se tokom pretrage određenog termina, nema potrebe da se prolazi kroz sam sadržaj knjige, tačnije stranu po stranu, već se na osnovu indeksa direktno pristupa strani koja sadrži određeni termin.

# MS SQL Server

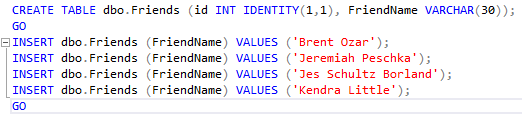
Microsoft SQL Server je jedan od najpopularnijih komercijalnih relacionih sistema za upraljvanje bazama podataka. Predstavlja softverski proizvod čiji je glavni zadatak da smešta i pribavlja podatke na zahtev drugih softverskih aplikacija. Skladište u kome se podaci smeštaju organizovan je kao skup tabela sa kolonama za raziličite tipove podataka, uključujući i primitivne tipove kao što su Integer, Float, Decimal, Char, Varchar, Binary,... Takođe, omogućava definisanje i upotrebu korisnički definisanih složenih tipova.

## 3.1. Organizacija skladišta

Svaka SQL Server baza podataka ima dve datoteke operativnog sistema: datoteke podataka (Data files) i datoteke evidencije (Log files). Datoteke podataka sadrže podatke i objekte, kao što su trabele, pogledi, uskladištene procedure i indekse. One se mogu podeliti na dva tipa, primarne i sekundarne datoteke. Primarne datoteke sadrže informacije o pokretanju baze i ukuzaju na ostale datoteke u bazi. Postoji samo jedna takva datoteka u bazi čija ekstenzija je .mdf. Sekundarne datoteke su izborne datoteke koje korisnik definiše i koje se koriste za širenje podataka među tiskovima, stavljanjem svake datoteke nma drugi disk. Ekstenzija ovih datoteka je .ndf. Datoteke evidencije sadrže informacije potrebne za oporavak svih transakcija u bazi podataka, i u svakoj bazi mora popstojati najmanje jedna takva datoteka čija je ekstenzija .ldf. Pored ovih datoteka operativnog sistema, postoje i grupe datoteka (Filegroups) koje rade kao logički kontejner za datoteke podataka i koji može da sadrži više datoteka podataka.

## Primer organizacije

U ovom odeljku ćemo kroz konkretan primer videti kreiranje baze i kako je ona organizovana u datoteke. Kreiraćemo tabelu *Friends* sa 4 podatka.

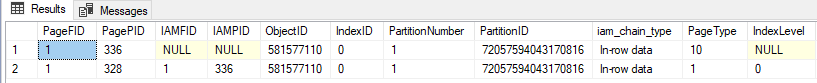


*Slika 1. - Komanda za kreiranje tabele i 4 podatka*

Vrlo je lako zaviriti u unutršnjost SQL Server baze podataka, u bajtove koji čine bazu podataka. Može se koristiti jedna od tri DBCC komandi: DBCCTRACEON, DBCCPAGE i DBCCIND. Microsoft poslednje dve nije zvanično nije dokumentovao ali ih ljudi veoma široko koriste.

DBCCIND nam pruža relevantne ID-ove stranice za bilo koji objekt u bazi podataka, a onda nam DBCCPAGE pruža mogućnost da istražimo šta se skladišti na disku na tim određenim stranicama. DBCCIND I DBCCPAGE su apsolutno sigurne za upotrebu. Ako izvršimo sledeću komandu, dobijamo listu stranica smeštenih u našoj bazi:

*DBCC IND('Sample', 'Friends', -1);*



*Slika 2. - Podaci smešteni na SQL Serveru*

Ove stranice su najmanja jedinica za skladištenje u memoriji i na disku. Kada upišemo prvi red u tabelu, SQL Server dodeljuje stranicu id 8KB za čuvanje tog reda a možda i još nekoliko u zavisnosti od veličine naših podataka. U ovom primeru, svaki red je mali pa ih možemo smestiti na jednu stranicu. Da imamo veće redove oni bi zauzimali više stranica, npr ako dodamo polje VARCHAR(MAX) i napunimo ga, ono će se prostirati na više strana.

Ako isključimo SQL Server i ponovo uključimo, pokrenemo komandu:

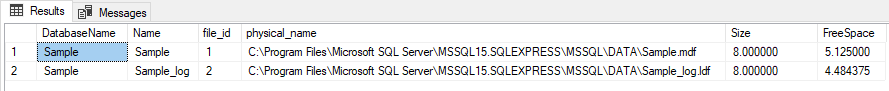
*SELECT \* FROM dbo.Friends WHERE FriendName = 'Brent Ozar';*

SQL Server bi proverio na kojoj stranici je naša tabela, zatim učitao čitavu 8KB stranicu sa diska i keširao tu stranicu u memoriju. Treba se naglasiti da se stranice podjednako čuvaju kako na disku tako i u memoriji. Ako se koristi kompresija podataka, podaci se neće eksportovati sa stranice sve dok ih nije potrebno ponovo učitati.

Ako ažuriramo bazu dodavanjem novog reda sledećom komandom:

*INSERT dbo.Friends (FriendName) VALUES ('Lady Gaga');*

Tako nastaje datoteka evidencije, sekvencijalni zapis onoga što smo uradili nad podacima. Kada napravimo novi, ažuriramo ili izbrišemo redove iz tabele, SQL Server prvo upiše tu aktivnost u datoteku evidencije.



*Slika 3. - Datoteke SQL Servera*

# Zaključak

Prostor za skladištenje dodeljen bazi podataka podeljen je na stranice s rednim brojem, svaka veličine 8KB. Stranica je osnovna ulazno-izlazna jedinica za operacije SQL Servera, označena je zaglavljem od 96 bajtova u kojem se čuvaju metapodaci o stranici uključujući broj stranice, tip stranice, slobodni prostor na stranici i ID objekta koji je vlasnik. Tip stranice definiše podatke sadržane na stranici. Iako je stranica osnovna jedinica, prostorom se zapravo upravlja opsegom koji se sastoji od 8 stranica. Objekat u bazi može da obuhvati svih 8 stranica u opsegu ili da deli udeo sa još 7 objekta. Red u tabeli baze ne može se prostirati na više stranica, pa je zbog toga ogrtaničen na 8KB. Međutim, ako podaci prelaze 8KB a red je tipa VARCHAR ili sadrži binarne podatke, podaci u tim kolonama se premeštaju na novu stranicu ( ili na sekvencu stranica koja se naziva alokacijska jedinica) i zamenjuju pokazivačem na podatke.

# Literatura

1. Ramez Elmasri and Shamkant Navathe. 2010. Fundamentals of Database Systems (6th. ed.). Addison-Wesley Publishing Company, USA.
2. Raghu Ramakrishnan and Johannes Gehrke. 2002. Database Management Systems (3rd. ed.). McGraw-Hill, Inc., USA.
3. Kalen Delaney, Paul S. Randal, Kimberly L. Tripp, Conor Cunningham, Adam Machanic, and Ben Nevarez. 2009. Microsoft SQL Server 2008 Internals. Microsoft Press
4. <https://docs.microsoft.com/en-us/sql/relational-databases/databases/database-files-and-filegroups?view=sql-server-ver15>
5. <https://en.wikipedia.org/wiki/Microsoft_SQL_Server>
6. <https://www.red-gate.com/simple-talk/sql/database-administration/sql-server-storage-internals-101/>
7. <https://www.brentozar.com/archive/2013/02/how-does-sql-server-store-data/>
8. <https://www.mssqltips.com/sqlservertip/4345/understanding-how-sql-server-stores-data-in-data-files/>